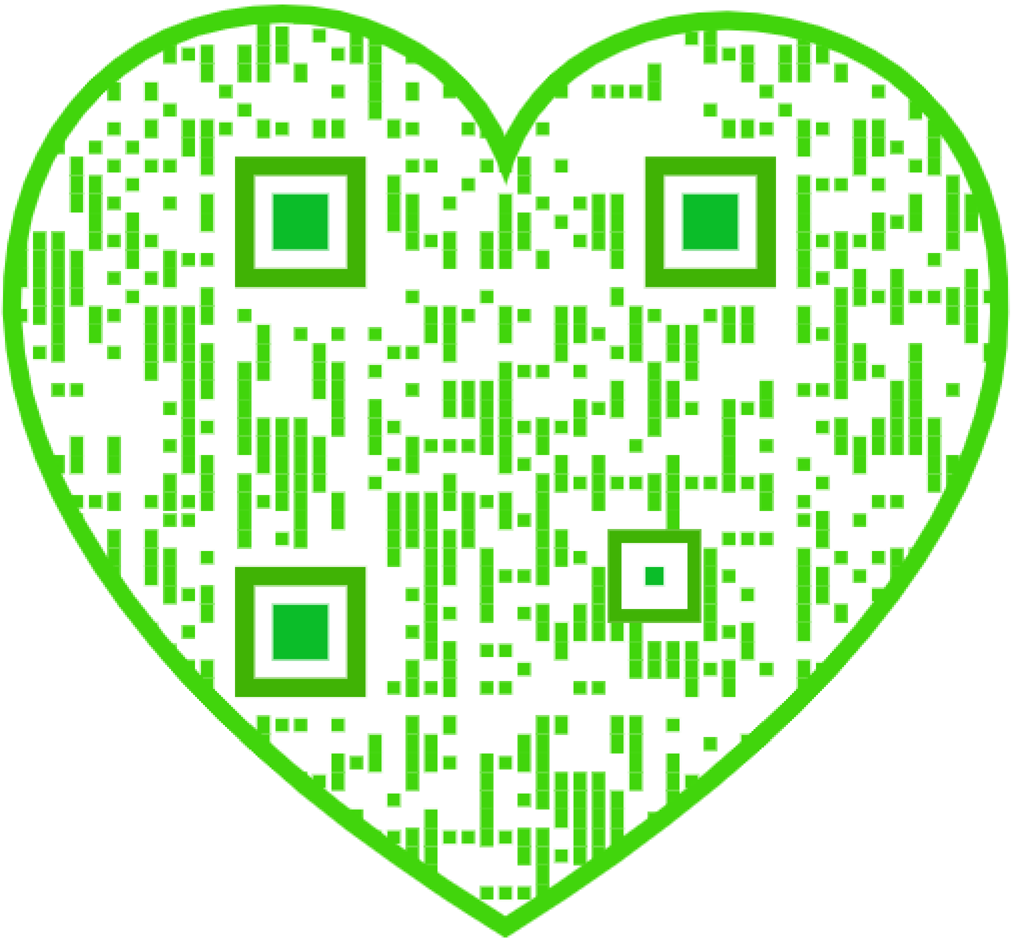
Master in Artificial Intelligence

Data Collection & Preprocessing VI



Purpose

The purpose of the section is to help you learn how to collect and preprocess data to become a Successful Artificial Intelligence (AI) Engineer

At the end of this lecture, you will learn the following

- **An example of gathering relevant data from various sources, ensure its quality, and preprocess it to make it suitable for analysis and modeling**

Bag-of-Words (BoW) representation

TF-IDF (Term Frequency-Inverse Document Frequency)

Word embeddings

Loading Pre-trained Word Embeddings

↓

Mapping Words to Embeddings

↓

Constructing Sentence Embeddings

How pre-trained word embeddings like Word2Vec or GloVe used to represent words as dense vectors in a continuous vector space-Example

Reviews in our dataset as the example

- "The product is great and works well."
- "I am satisfied with my purchase."
- "This product is terrible and does not work."

Tokenized each review into individual words and mapped each word

Then aggregated the word embeddings for each review to obtain a dense vector representation

How did the dense vector representations obtained from the pre-trained word embeddings look like in the above example

In the above example, we had pre-trained Word2Vec embeddings available for the words in the reviews. Each word in the vocabulary was mapped to a dense vector in a continuous vector space based on its semantic meaning captured from the large corpus used for training the Word2Vec model.

Here's how the dense vector representations obtained from the pre-trained Word2Vec embeddings looked like for the example reviews: following Word2Vec embeddings for some words were obtained:

- **"product": [0.2, -0.3, 0.1, ...]**
- **"great": [0.5, 0.2, -0.1, ...]**
- **"works": [0.3, 0.1, 0.4, ...]**
- **"well": [0.4, -0.2, 0.3, ...]**
- **"satisfied": [0.1, 0.4, -0.3, ...]**
- **"purchase": [0.2, 0.3, 0.2, ...]**
- **"terrible": [-0.3, 0.1, -0.4, ...]**
- **"does": [-0.1, -0.2, 0.5, ...]**
- **"not": [-0.2, 0.1, -0.3, ...]**

Review 1: "The product is great and works well."

Average Word2Vec embeddings

[0.2, -0.3, 0.1, ...] (product) +
[0.5, 0.2, -0.1, ...] (great) +
[0.3, 0.1, 0.4, ...] (works) +
[0.4, -0.2, 0.3, ...] (well)

Resulting dense vector representation for Review 1

[(0.2 + 0.5 + 0.3 + 0.4) / 4, (-0.3 + 0.2 + 0.1 - 0.2) / 4, (0.1 - 0.1 + 0.4 + 0.3) / 4, ...]

Review 2: "I am satisfied with my purchase."

Average Word2Vec embeddings

[0.1, 0.4, -0.3, ...] (satisfied) +
[0.2, 0.3, 0.2, ...] (purchase)

Resulting dense vector representation for Review 2

[(0.1 + 0.2) / 2, (0.4 + 0.3) / 2, (-0.3 + 0.2) / 2, ...]

Review 3: "This product is terrible and does not work."

Average Word2Vec embeddings

[0.2, -0.3, 0.1, ...] (product) +
[-0.3, 0.1, -0.4, ...] (terrible) +
[-0.1, -0.2, 0.5, ...] (does) +
[-0.2, 0.1, -0.3, ...] (not) +
[0.3, 0.1, 0.4, ...] (works)

Resulting dense vector representation for Review 3

[(0.2 - 0.3 - 0.1 - 0.2 + 0.3) / 5, (-0.3 + 0.1 - 0.2 + 0.1) / 5, (0.1 - 0.4 + 0.5 - 0.3) / 5, ...]

Usage in Modeling

Dense vector representations obtained from the pre-trained word embeddings

Make accurate predictions

These embeddings captured semantic similarities between words

Allowing the model to

Learn meaningful representations of the text data and

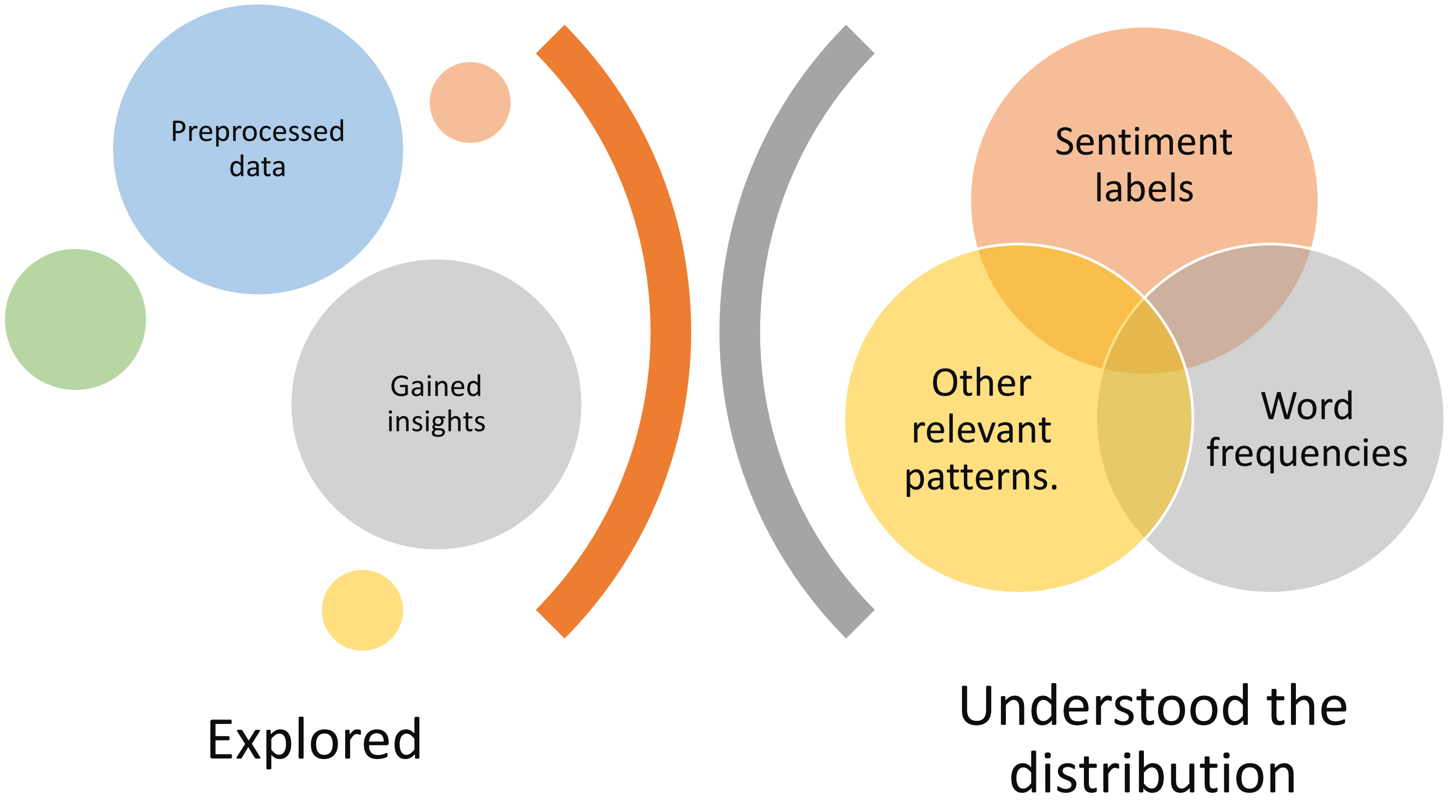
Input features for machine learning models

Bag-of-Words (BoW) representation

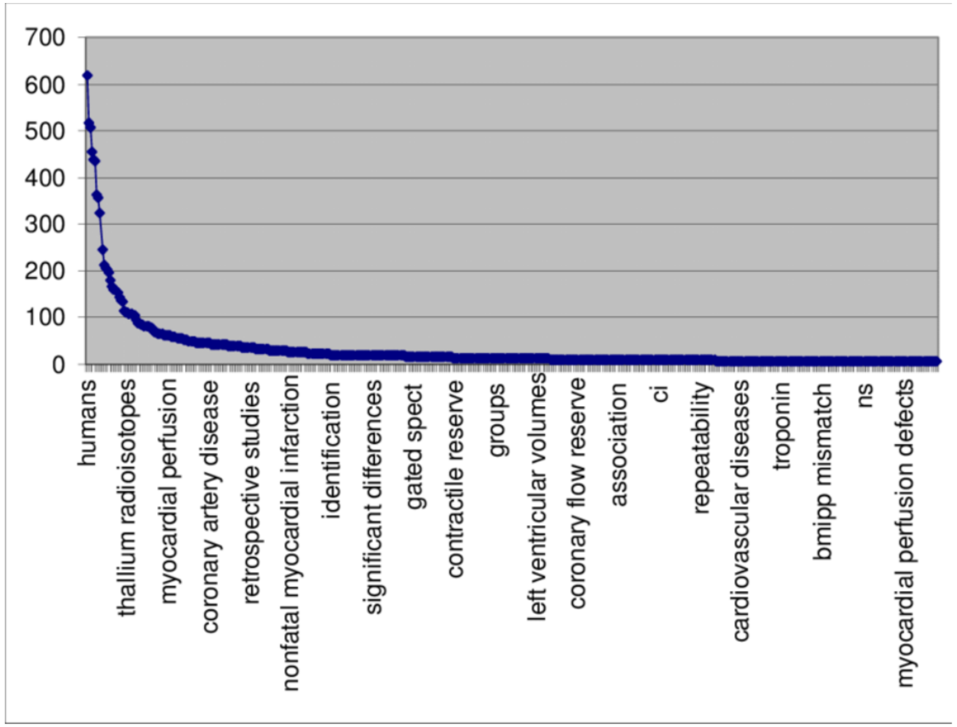
TF-IDF (Term Frequency-Inverse Document Frequency)

Word embeddings

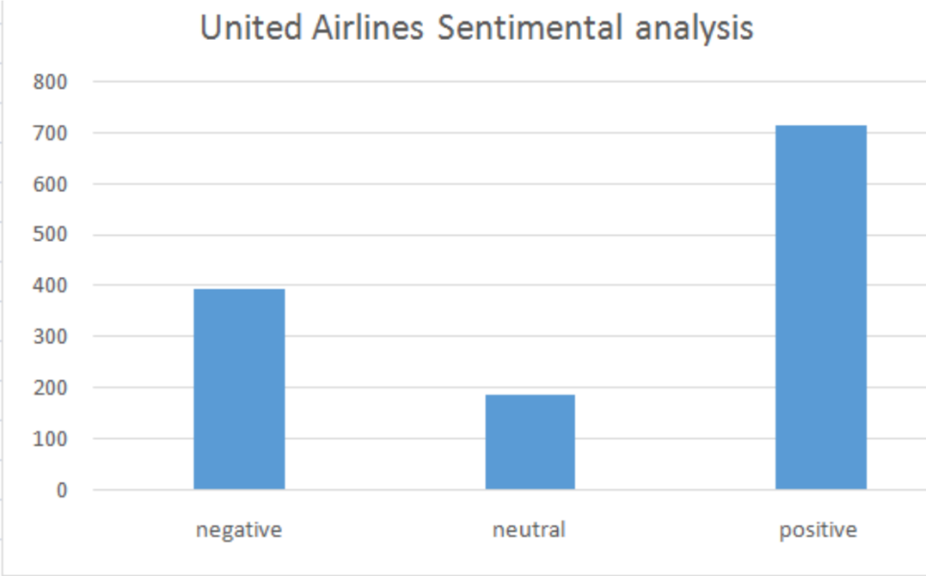
Exploratory Data Analysis (EDA)



Exploratory Data Analysis (EDA)



Top 500 Word Frequency Distribution from extracted terms



Histogram for sentimental analysis Fig. 6. Pie chart for sentimental analysis

How to collect and preprocess data- An Example

Gather relevant data

Ensure its quality

Preprocess it

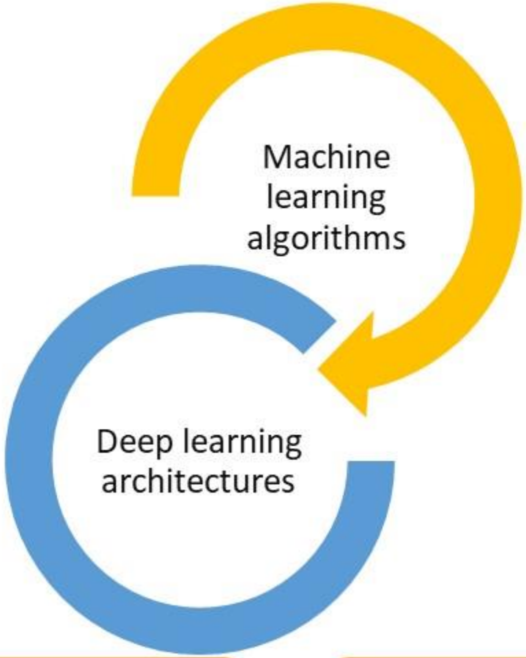
Make it suitable for analysis and modeling.

Algorithm Selection and Development

Research

Select

Develop



Problem at hand

Available data

Master in Artificial Intelligence

Data Collection & Preprocessing VI